REMARKS

Claims 1, 2, 4-14 and 27-42 are pending in this case. Reconsideration of the application is respectfully requested in light of the above amendments and the following remarks.

This amendment is responsive to the Office Action mailed January 29, 2003. In that Office Action, claims 1-26 were examined; claims 1, 5, 12, and 22 were rejected under 35 U.S.C. § 112, second paragraph, as being indefinite; and claims 1-26 were rejected under 35 U.S.C. § 102(e) as being unpatentable over Johnston, Jr. et al., (U.S. Pat. No. 6,104,391, hereinafter "Johnston").

Specification and Claim Amendments

The specification has been amended to correct a typographical error. Claims 1, 5 and 12 have been amended to correct informalities noted by the Examiner. Claim 1 has been further amended to include the limitations of claim 3 and claim 3 has been canceled. Claims 15-26 have been canceled. New claims 27-42 have been added. All canceled claims are canceled without prejudice and the Applicants reserve the right to prosecute the canceled claims in a separate continuing application at a later date.

Rejections Under 35 U.S.C. § 112, Second Paragraph

Claims 1, 5, and 12 have been amended to correct the antecedent basis as requested by the Examiner. Claim 22 has been canceled, so the rejection of it is moot.

Rejections Under 35 U.S.C. § 102(e)

The Examiner rejected claims 1-26 under 35 U.S.C. § 102(e) as being unpatentable over Johnston, stating that "Johnston discloses an analogous system and method" (Page 5). The Applicants traverse this rejection and disagree with the Examiner's characterization of Johnston. The Applicants agree that, as far as the perception of a user is concerned, the methods of Johnston and the present invention may achieve similar results and that Johnston's language is similar to that of the Applicants. However, the present invention uses methods and systems that are entirely different from those disclosed by Johnston.

Johnston presents two embodiments of a method for using an appearance management layer that gives users the ability to customize the appearance of a desktop by switching between themes. In the first embodiment, to switch themes the appearance management layer switches the drawing procedures used by the applications by causing all utilities that use drawing procedures to switch the drawing procedures that they call.

> According to certain exemplary embodiments, all of the utilities which support switchable drawing procedures will be called to "disconnect" all of the drawing procedures for each of the interface objects supported by that particular utility. In essence, this amounts to sending a dispose message to the drawing procedure for each and every utility object element currently in existence. The utility then is called to swap pointers 44 to the drawing procedures. For example, if window drawing procedure A is being replaced by window drawing procedure B, the window drawing utility will be asked to replace all of its references to procedure A with references to procedure B. This process will occur for each drawing procedure that is switched out. Johnston, Col. 6, lines 40-53.

The second embodiment of Johnston is, rather than switching the pointers in the utilities, to provide new data that the utilities then provide to parametric drawing procedures. Each theme is provided with its own data structures and the appearance management layer, upon a theme change, updates the data structures.

13

According to other exemplary embodiments of the present invention, these drawing procedures can be data driven so as to allow each procedure to be able to support a wide variety of appearances and behaviors without modifying the code of the procedure itself. In this way themes can be switched without requiring that the drawing procedure code be switched. Each theme provides its own data structures which are supplied to the parametric drawing procedure. Johnston, Col. 6, lines 57-65.

Both embodiments disclose a method whereby, upon a theme change, the management layer changes the drawing procedures or theme data structures to those of the new theme. In both embodiments, something is changed in the utilities or the appearance manager to effect the theme change. In the first embodiment, the utilities' pointers are replaced by new pointers. In the second embodiment, old data structures are replaced by new data structures. Johnston dedicates FIG. 13 to the steps necessary to "open the new theme file" by creating a "new resource chain" - all of which are required to "set" the new theme. See e.g., Col. 5, lines 30-55. Thus, Johnston's appearance management layer, after effectuating the theme change, is not required until the next theme change. Note also that the fundamental concept of Johnston is that the applications 38 of Johnston are not affected by the theme changes; "applications no longer need to have a priori knowledge of the patterns or colors used." Johnston Col. 5, lines 57-58.

The Examiner rejected claim 3, now amended claim 1, equating "the graphical component library to "item 38 or client" (Office Action Page 6, line 4) and Applicants' theme handle with items "46, 50, 70 and 72" (Office Action Page 6, line 13). These analogies are incorrect. First, Johnston's item 38 is an application, which by Johnston's own definition does not receive information from Johnston's appearance management layer 40. Johnston's applications 38 do not request a theme handle nor do they receive a theme handle. Johnston's

14

applications 38, in fact, are totally unaware of the existence of the appearance management layer 38.

With respect to the Applicants' theme handle, the Applicants define it on page 19, line 16 of the application as "a reference to an internal structure detailing various information and properties of the object." Nowhere does Johnston disclose an application 38 requesting this type of information from the appearance management layer 40. In Johnston, when a theme is set the appearance management layer 40 implements all the changes necessary. Johnston does not disclose any elements of the computer actively requesting anything from the appearance management layer 40 other than rendering requests.

These are just two of the differences between Johnston and the present invention. There are many others, such as that the appearance management layer 40 is not analogous to the theme manager or the appearance manager and the graphical component library 212 is not analogous to any distinct element presented in Johnston. These differences arise because the architecture of Johnston is fundamentally different than the Applicants' architecture as shown in FIG. 1.

For the reasons above and others that are now readily apparent in light of the above discussion, the Applicants believe that amended claim 1 is in a condition for allowance. Therefore, the Applicants respectfully request that the Examiner withdraw his rejection and find amended claim 1, and its pending dependent, i.e., claims 2 and 4-9, in allowable form.

Turning now to the Examiner's rejection of claim 10, The Examiner will find that the above arguments apply as well. The Applicants' theme handle cannot be equated to "the functions of device 46/50, 70 & 72" as the Examiner did on Page 8, lines 5-6 of the Office Action. Johnston's 46 refers to a library of drawing procedures. Johnston's 50 refers to a theme switching utility that changes pointers in utilities 42. Johnston's 70 is a library of theme

definitions. Johnston's 72 is a theme's resource chain. Johnston simply does not disclose "a reference to an internal structure detailing various information and properties of the object" because the architecture of Johnston does not require the use of theme handle. In Johnston, the appearance management layer knows what theme is active and does not need to pass this information to any other component in the architecture. Therefore, Johnston does not anticipate requesting a theme handle because Johnston does not need one or use one. In Johnston, nothing external to the appearance management layer knows or cares what theme is currently being used.

In addition, the Examiner again equates the graphical component library with an application 38, even though Johnston explicitly says that the application 38 is totally independent of the theme management performed by the appearance management layer 40. Applicants' claim 10, on the other hand, requires that the graphical component library request and receive a theme handle. Johnston does not disclose or teach requesting and receiving theme handles from an appearance manager and, therefore, clearly does not anticipate this claim.

For the reasons above and others that are now readily apparent in light of the above discussion, the Applicants believe that claim 10 is in a condition for allowance. Therefore, the Applicants respectfully request that the Examiner withdraw his rejection and find claim 10, and its pending dependent claims, i.e., 11-14, in allowable form.

The Applicants, by this amendment, have added new claims 27-42. Claims 27-35 include several elements in FIG. 1 such as the fusion 214 described in the application on pages 15-16, pages 27-28 with reference to FIG. 4, and elsewhere. The fusion process determines if the requested graphical component is a theme-aware component. Page 28, lines 1-2. Johnston does not teach or imply theme-aware components, nor a fusion process that determines if a graphical component is a theme-aware component. Remember that Johnston is a static redefinition of

either pointers or data. No dynamic determinations are made in Johnston as each graphic rendering call has only one pointer or only one set of data. There is never any need for a dynamic selection between two or more possible renderings in Johnston.

In addition, claim 27 further requires that if the graphical component is not theme-aware the graphical component is rendered by a system graphical component library. On the other hand, theme-aware graphical components are rendered by a theme graphical component library using a theme manager. Johnston does not teach or disclose a system graphical component library separate and distinct from a theme graphical component library. Furthermore, as discussed above Johnston does not disclose a theme graphical component library that calls a theme manager when rendering theme-aware graphical components. For these and other reasons the Applicants believe that new claims 27-35 are distinguishable over Johnston and, therefore, should be allowed by the Examiner.

New claims 36-39 relate to a method for using of theme handles to assist in setting a theme. Johnston does not disclose any similar method for setting a theme, as in Johnston it is all a matter of the appearance management layer changing pointers or data. Furthermore, as discussed above, Johnston does not teach or disclose a fusion process or theme-handles in any way. Therefore, the Applicants believe that claims 36-39 are also distinguishable over Johnston and should be allowed.

Lastly, the Applicants have added new claims 40-42 which relate to the architecture of the Applicants' invention as shown in FIG. 1. As the architecture and functional elements of Johnston are quite different from that of the present invention, the Applicants believe that claims 40-42 are also distinguishable over Johnston. For instance, the Applicants claim a fusion process which is not disclosed or taught by Johnston. The Applicants further claim a theme graphical